UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/870,614	05/31/2001	Scott J. Broussard	AUS920010265US1	1779

48916 7590 06/09/2010

Greg Goshorn, P.C.
9600 Escarpment
Suite 745-9
AUSTIN, TX 78749

EXAMINER	
BONSHOCK, DENNIS G	

ART UNIT	PAPER NUMBER
2173	

MAIL DATE	DELIVERY MODE
06/09/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/870,614
Filing Date: May 31, 2001
Appellant(s): BROUSSARD, SCOTT J.

Gregory K. Goshorn (Reg. No.: 44,721)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 3-5-2010 and supplemental appeal brief filed 03-30-2010 appealing from the Office action mailed 10-05-2009.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The following are the related appeals, interferences, and judicial proceedings known to the examiner which may be related to, directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal:

09/870,613

09/870,615

09/870,620

09/870,621

09/870,622

09/870,624

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application:

Claims 1-22 are currently pending and have been finally rejected.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

4,951,229	DiNicola et al.	8-1990
5,838,336	Ross	11-1998
6,522,341	Nagata	2-2003

Sun Microsystems, Mixing Heavy and Light components, 2/98, volumn 3, no.4, swing version 1.0

Sun Microsystems, Introducing Swing, 2.98, volumne 3, no. 4, swing version 1.0

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1, 2, 5, 6, 20, and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over DiNicola et al., Patent #4,951,229, hereinafter DiNicola, Ross, Patent Number: 5,838,336, and Nagata, Patent #6,522,341.

With regard to claim 1, which teaches a display system, comprising: a display, DiNicola teaches, in column 2, line 68, the use of a display. With regard to claim 1 further teaching a display buffer coupled to the display, DiNicola teaches, in column 2, line 59 through column 3, line 16, the use of several different buffers used in the display of an image. With regard to claim 1, further teaching a processor adapted to execute an application program; DiNicola teaches, in column 3, lines 7-10, the use of a processor in the application program. With regard to claim 1, further teaching producing images on the display where the images are either in a first mode by forwarding in sequence to the display or in a second mode, compiled as a combination image of at least one image drawn over another image, DiNicola teaches, in column 2, line 59 through column 3, line 16, specifically column 3, lines 10-13, a system that can be configured to either send images to the display separately or to combine two or more of the images and send them as a composite display image. With regard to claim 1, further teaching presenting

the image to the buffer before forwarding, DiNicola teaches, in column 3, lines 62-68 and column 5, lines 18-32, buffering the images before sending to the display.

DiNicola, however, doesn't specifically teach a second mode buffering the combination image prior to display. Ross teaches a system for combining images for display via a frame buffer (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2), similar to that of DiNicola, but further teaches two distinct modes where in the "hardware cursor mode"/"overlay mode" each image is transmitted din sequence to the display screen (ie: the on-screen memory is sent then the cursor memory is sent overlaying the on-screen memory) and alternately in a "normal mode" each two images a are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola and Ross before him at the time the invention was made to modify the image combination system of DiNicola to include the combinational mode with pre buffering of Ross. One would have been motivated to make such a combination because this allows for one image to be transmitted to the display rather than two (easing the transmission beneficial in non-video related image processing).

DiNicola and Ross teach displaying images separately on a display and combining the into a composite image and placing it in frame buffer for display (see column 2, lines 25-39 and column 2, line 66 through column 3, line 13), however, don't specifically teach combining the two images into one image and buffering this combined image prior to display. Nagata teaches a system in which multiple images are

combined for display (see abstract), similar to that of DiNicola and Ross, but further teaches a the steps of a mixer mixing the two images and then transmitting the combined image to a buffer prior to displaying the image (see column 2, line 15 through column 3, line 4 and in figures 1and 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, and Nagata before him at the time the invention was made to modify the image combination system of DiNicola and Ross to include the combination of images followed by buffering the combined image, followed by display, as is done in Nagata. One would have been motivated to make such a combination because this allows for reuse of the displayed image for further additions to the display space (as would be the case in animation / video environments).

With regard to claim 2, which teaches the application program disables or enables buffering of the images by configuring the processor to execute in either or the first or second mode, DiNicola teaches, in column 2, line 59 through column 3, line 16, specifically column 3, lines 10-13, a system that can be configured to either send images to the display separately or to combine two or more of the images and send them as a composite display image, there for providing them with the optional intermediate buffer (see column 5, lines 18-27).

With regard to claim 5, which teaches a computer-readable memory, comprising: an operating system, DiNicola teaches, in column 3, line 50 an operating system. With regard to claim 5, further teaching an application program running on code compatible with the operating system, DiNicola teaches, in column 1, lines 6-15, an application program running on code compatible with the operating system. With regard to claim 5,

further teaching a software component invoked by an application program that produces images on the display where the images are either forwarded in sequence to the display or are compiled as a combination image of at least on image drawn over another image, DiNicola teaches, in column 2, line 59 through column 3, line 16, specifically column 3, lines 10-13, a system that can be configured to either send images to the display separately or to combine two or more of the images and send them as a composite display image. With regard to claim 5, further teaching the optional buffering of the sequence of images as a combination image before sending the combination image to the display, DiNicola teaches, in column 3, lines 62-68 and column 5, lines 18-32, an intermediate buffer that is not required, as a matter of efficiency, but mentioned in the reference.

DiNicola, however, doesn't specifically teach a second mode buffering the combination image prior to display. Ross teaches a system for combining images for display via a frame buffer (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2), similar to that of DiNicola, but further teaches two distinct modes where in the "hardware cursor mode"/"overlay mode" each image is transmitted din sequence to the display screen (ie: the on-screen memory is sent then the cursor memory is sent overlaying the on-screen memory) and alternately in a "normal mode" each two images a are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola and Ross before him at the time the invention was made to modify

the image combination system of DiNicola to include the combinational mode with pre buffering of Ross. One would have been motivated to make such a combination because this allows for one image to be transmitted to the display rather than two (easing the transmission beneficial in non-video related image processing).

DiNicola and Ross teach displaying images separately on a display and combining the into a composite image and placing it in frame buffer for display (see column 2, lines 25-39 and column 2, line 66 through column 3, line 13), however, don't specifically teach combining the two images into one image and buffering this combined image prior to display. Nagata teaches a system in which multiple images are combined for display (see abstract), similar to that of DiNicola and Ross, but further teaches a the steps of a mixer mixing the two images and then transmitting the combined image to a buffer prior to displaying the image (see column 2, line 15 through column 3, line 4 and in figures 1and 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, and Nagata before him at the time the invention was made to modify the image combination system of DiNicola and Ross to include the combination of images followed by buffering the combined image, followed by display, as is done in Nagata. One would have been motivated to make such a combination because this allows for reuse of the displayed image for further additions to the display space (as would be the case in animation / video environments).

With regard to claim 6, which teaches object code being part of a graphical user interface associated with the application program, DiNicola teaches, in column 3, lines 10-13, a graphical user interface associated with the application program.

With regard to claim 20, which teaches the processor executing in the first mode when the display is directly coupled to the processor, DiNicola teaches, in column 4, lines 31-53, column 5, lines 4-17, and figures 1 and 6, the processor being able to execute in two modes, one which provides the images as combined image and one which provides individual images to the display. DiNicola, Ross, and Nagata, however, doesn't state that the selection is made due to the relative location of the display. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, and Nagata before him at the time the invention was made to modify invention of DiNicola, Ross, and Nagata to make the selection based on the relative location of the display. One would have been motivated to make such a combination because remote displays require transmission over what is some times slow medium, which would be less efficient if a plurality of images were sent as opposed to a combined image.

With regard to claim 21, which teaches the processor executing in the second mode when the display is remotely coupled to the processor, DiNicola teaches, in column 4, lines 31-53, column 5, lines 4-17, and figures 1 and 6, the processor being able to execute in two modes, one which provides the images as combined image and one which provides individual images to the display. DiNicola, Ross, and Nagata, however, doesn't state that the selection is made due to the relative location of the

display. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, and Nagata before him at the time the invention was made to modify invention of DiNicola, Ross, and Nagata to make the selection based on the relative location of the display. One would have been motivated to make such a combination because remote displays require transmission over what is some times slow medium, which would be less efficient if a plurality of images were sent as opposed to a combined image.

Claims 3, 4, and 7-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over DiNicola, Ross, Patent Number: 5,838,336, Nagata, Patent #6,522,341, and Fowler, *Mixing Heavy and Light Components*.

With regard to claim 3, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13) and the use of 3 images layered on top of one another in a combination image (see column 5, lines 32-46), but does not teach images comprising frame, panel, and button images. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case, but further teaches the use of Frame, Panel, and Button images (see Z-order limitations (page 5), and Swing scroll pane (page 6)). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, and Nagata with the Frame, Panel, and Button

images of Fowler. One would have been motivated to make such a combination because these are elements implemented in Swing which is referred to in the specification on page 34, the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system, and because Frame, Panel, and Button images are images that are frequently transferred to displays.

With regard to claim 4, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), but does not teach the use of Java. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case, but further teaches the use Java see page 2, paragraph 4. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, and Nagata, to include the API system using Java of Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 7, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), but does not teach a software component comprising an API of code, which translates

between code within the application program and the operating system. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case, and further teaches the use Java see page 2, paragraph 4. It is inherently known in the art that Swing and AWT are application program interfaces, which are defined as a set of routines that translate between an application program and a computer's operating system (see Microsoft Computer Dictionary Fifth Edition, page 33). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, and Nagata, to include use of an API as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 8, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), but does not teach an API that emulates that of a second API based on a windows-based version of the operating system. Fowler teaches Mixing Swing and AWT in the same application program (see page 1, paragraph 2), as is taught in the specification of the case, and further teaches this being a windows based operating system (see the figure on page 7.). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Fowler before him at the time the invention

was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, and Nagata, to include use of two different APIs as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 9, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), but does not teach the use of a second API, that of AWT. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, and Nagata, to include use of two different APIs, including AWT as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 10, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), but does not teach the application program being written in Java. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is

taught in the specification of the case, and further teaches the use Java see page 2, paragraph 4. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, and Nagata, to include use of Java as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, where Swing is said to default to generating a combined image, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

Claims 11-13, 18, 19, and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over DiNicola, Sun Microsystems, *Introducing Swing,* hereinafter Sun, Ross, Patent Number: 5,838,336, and Nagata, Patent #6,522,341.

With regard to claims 11 and 18, DiNicola teaches the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), but does not teach the operating system comprising a Windows, Unix, or OS/2 computer operating system. Sun teaches Swing which is stated in the specification of the application to default to buffering it's output, similar to that of the DiNicola, Ross, Nagata, but further teaches on page 1, paragraph 4, running under operating systems such as Windows, Unix, and so on. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, and Sun

before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, Nagata, to include use of Java as did Sun. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, where Swing is said to default to generating a combined image, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 12, DiNicola teaches, in column 1, lines 6-15, an application program running on code compatible with the operating system, and a system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13), the optional buffering of the sequence of images as a combination image before sending the combination image to the display (see column 3, lines 62-68 and column 5, lines 18-32), but doesn't teach creating a graphical representation of the object using an interface independent of the operating system. Sun teaches Swing which is stated in the specification of the application to default to buffering it's output, similar to that of the DiNicola, but further teaches on page 1, paragraphs 4 and 5, swing being an API that can be independent of the operating system. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola and Sun before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola to include use of Java and it's cross platform component for Swing, as did Sun. One would have been motivated to make such a combination because Swing is referred

to in the specification on page 34, where Swing is said to default to generating a combined image, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

DiNicola and Sun, however, doesn't specifically teach a second mode buffering the combination image prior to display. Ross teaches a system for combining images for display via a frame buffer (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2), similar to that of DiNicola and Sun, but further teaches two distinct modes where in the "hardware cursor mode"/"overlay mode" each image is transmitted din sequence to the display screen (ie: the on-screen memory is sent then the cursor memory is sent overlaying the on-screen memory) and alternately in a "normal mode" each two images a are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Sun and Ross before him at the time the invention was made to modify the image combination system of DiNicola and Sun to include the combinational mode with pre buffering of Ross. One would have been motivated to make such a combination because this allows for one image to be transmitted to the display rather than two (easing the transmission beneficial in non-video related image processing).

DiNicola, Sun and Ross, however, don't specifically teach combining the two images into one image and buffering this combined image prior to display. Nagata

teaches a system in which multiple images are combined for display (see abstract), similar to that of DiNicola, Sun and Ross, but further teaches a the steps of a mixer mixing the two images and then transmitting the combined image to a buffer prior to displaying the image (see column 2, line 15 through column 3, line 4 and in figures 1and 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Sun, Ross, and Nagata before him at the time the invention was made to modify the image combination system of DiNicola, Sun and Ross to include the combination of images followed by buffering the combined image, followed by display, as is done in Nagata. One would have been motivated to make such a combination because this allows for reuse of the displayed image for further additions to the display space (as would be the case in animation / video environments).

With regard to claim 13, which teaches object code being part of a graphical user interface associated with the application program, DiNicola further teaches, in column 3, lines 10-13, a graphical user interface associated with the application program.

With regard to claim 19, DiNicola teaches a computer-readable storage device, comprising: an application program running under an operating system (see column 1, lines 6-15), an object created at runtime by the application program (see column 2, line 59 through column 3, lines 16; specifically column 3, lines 10-13), and creating a graphical representation of the object; enabling or disable buffering of the graphical representation of the object to a memory storage area prior to displaying the graphical representation, as directed by the application program (see column 2, line 59 through

column 3, lines 16; specifically column 3, lines 10-13, and column 5, lines 18-32). DiNicola, however, doesn't teach a windows-based operating system or an interface independent of the operating system. Sun teaches Swing that is stated in the specification, as defaulting to buffering it's output, similar to that of DiNicola, but further teaches a windows-based operating system (see page 1, paragraph 4), and an interface that is independent of the operating system (see page 1, paragraphs 4 and 5). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola and Sun before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola to include use of Java and it's cross platform component for Swing, as did Sun. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, where Swing is said to default to generating a combined image, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

DiNicola and Sun, however, doesn't specifically teach a second mode buffering the combination image prior to display. Ross teaches a system for combining images for display via a frame buffer (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2), similar to that of DiNicola and Sun, but further teaches two distinct modes where in the "hardware cursor mode"/"overlay mode" each image is transmitted din sequence to the display screen (ie: the on-screen memory is sent then the cursor memory is sent overlaying the on-screen memory) and alternately

in a "normal mode" each two images a are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Sun and Ross before him at the time the invention was made to modify the image combination system of DiNicola and Sun to include the combinational mode with pre buffering of Ross. One would have been motivated to make such a combination because this allows for one image to be transmitted to the display rather than two (easing the transmission beneficial in non-video related image processing).

DiNicola, Sun and Ross, however, don't specifically teach combining the two images into one image and buffering this combined image prior to display. Nagata teaches a system in which multiple images are combined for display (see abstract), similar to that of DiNicola, Sun and Ross, but further teaches a the steps of a mixer mixing the two images and then transmitting the combined image to a buffer prior to displaying the image (see column 2, line 15 through column 3, line 4 and in figures 1and 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Sun, Ross, and Nagata before him at the time the invention was made to modify the image combination system of DiNicola, Sun and Ross to include the combination of images followed by buffering the combined image, followed by display, as is done in Nagata. One would have been motivated to make such a combination because this allows for reuse of the displayed image for further additions to the display space (as would be the case in animation / video environments).

With regard to claim 22, determining if the application program is operating in remote or direct mode, and creating a peer component to enable or disable buffering of the graphical representation of the object based on the determination made by the software component, DiNicola teaches, in column 4, lines 31-53, column 5, lines 4-17, and figures 1 and 6, the processor being able to execute in two modes, as selected by the selector [32], one which provides the images as combined image and one which provides individual images to the display. DiNicola, Sun and Ross, however, don't state that the selection is made due to the relative location of the display. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Sun and Ross before him at the time the invention was made to modify invention of DiNicola, Sun and Ross to make the selection based on the relative location of the display. One would have been motivated to make such a combination because remote displays require transmission over what is some times slow medium, which would be less efficient if a plurality of images were sent as opposed to a combined image.

Claims 14-17 are rejected under 35 U.S.C. 103(a) as being unpatentable over DiNicola, Ross, Nagata, Fowler, and Sun.

With regard to claim 14, DiNicola, Ross, Nagata, and Sun teach the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13) and the use of 3 images layer on top of one another in a combination image (see column 5, lines 32-46), but does not teach a software

component comprising an API of code which translates between code within the application program and the operating system. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case, and further teaches the use Java see page 2, paragraph 4. It is inherently known in the art that Swing and AWT are application program interfaces, which are defined as a set of routines that translate between an application program and a computer's operating system (see Microsoft Computer Dictionary Fifth Edition, page 33). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata, Sun, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, Nagata, and Sun to include use of an API as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 15, DiNicola, Ross, Nagata, and Sun teach the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13) and the use of 3 images layer on top of one another in a combination image (see column 5, lines 32-46), but does not teach an API that emulates that of a second API based on a windows based operating system. Fowler teaches Mixing Swing and AWT in the same application program (see page 1, paragraph 2), as is taught in the specification of the case, and further teaches this being

a windows based operating system (see the figure on page 7). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata,, Sun, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, Nagata, and Sun to include use of two different APIs as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 16, DiNicola, Ross, Nagata, and Sun teach the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13) and the use of 3 images layer on top of one another in a combination image (see column 5, lines 32-46), but does not teach the use of a second API, that of AWT. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata,, Sun, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, Nagata, and Sun to include use of two different APIs, including AWT as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, and because the use of Java would

allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

With regard to claim 17, DiNicola, Ross, Nagata, and Sun teach the system that either transmits images to the display sequentially or as a combination image (see column 3, lines 10-13) and the use of 3 images layer on top of one another in a combination image (see column 5, lines 32-46), but does not teach the application program being written in Java. Fowler teaches Mixing Swing an AWT in the same application program (see page 1, paragraph 2) as is taught in the specification of the case, and further teaches the use Java see page 2, paragraph 4. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, Nagata,, Sun, and Fowler before him at the time the invention was made to modify the system of transmitting images either in sequence or as a combined image of DiNicola, Ross, Nagata, and Sun to include use of Java as did Fowler. One would have been motivated to make such a combination because Swing is referred to in the specification on page 34, where Swing is said to default to generating a combined image, and because the use of Java would allow for portability of the image display system, this would prove useful if the remote system was run on a different operating system.

(10) Response to Argument

Claims 1-22*:*

With respect to the arguments directed at the independent claims including Claims 1, 5, 12, and19 the Appellant's arguments are focused on the limitations regarding

the existence of a buffer between the combined image and the display. More specifically, as stated from representative Claim 1, the limitation argued is:

"

Images are compiled as a combination image of at least one of said image drawn over at least another of said images and presented to the buffer before being forwarded to the display

"

Since the interpretation of the limitation is the basis for the arguments, the Examiner's interpretation is now given. The claim, as interpreted by the examiner, pertains to a system in which two images are combined and then buffered prior to being displayed. As stated in the eighth paragraph of MPEP 2101[R2].II.C.,

"*Office personnel are to give claims their broadest reasonable interpretation in light of the supporting disclosure.* In re Morris, 127 F.3d 1048, 1054-55, 44 USPQ2d 1023,1027-28 (Fed. Cir. 1997)."

Based on the interpretation of the claim limitations being argued, the Examiner will now explain how the teachings of the DiNicola, Ross, and Nagata references are within the scope of these limitations.

DiNicola teaches, in column 2, line 59 through column 3, line 16, the use of several different buffers used in the display of an image. DiNicola teaches, in column 2, line 59 through column 3, line 16, specifically column 3, lines 10-13, a system that can be configured to either send images to the display separately or to combine two or more of the images and send them as a composite display image. DiNicola teaches, in column 3, lines 62-68 and column 5, lines 18-32, buffering the images before sending to the display, as the bit planes 24, 26, 28, of DiNicola are described as "memory buffers" (see column 5, lines 4-16 and column 6, lines 18-20 and lines 30-32).

DiNicola is further supplemented by Ross who further teaches a system for combining images for display via a frame buffer (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2), similar to that of DiNicola, but further teaches two distinct modes where in the "hardware cursor mode"/"overlay mode" each image is transmitted in sequence to the display screen (ie: the on-screen memory is sent, then the cursor memory is sent overlaying the on-screen memory) and alternately in a "normal mode" each of the two images are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2). In other words Ross has two images in memory (1) the on-screen memory [18]; and (2) the cursor memory [20]; and the display memory [30], which is described as a frame buffer that stores the combined on-screen memory [18] and cursor memory [20] (see column 3, lines 50-57), as does a conventional display memory storing the image (one image frame) that is to be

displayed on the screen. It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola and Ross before him at the time the invention was made to modify the image combination system of DiNicola to include the combinational mode with pre buffering of Ross. One would have been motivated to make such a combination because this allows for one image to be transmitted to the display rather than two (easing the transmission beneficial in non-video related image processing).

DiNicola and Ross teach displaying images separately on a display and alternately combining them and placing it in frame buffer for display (see column 2, lines 25-39 and column 2, line 66 through column 3, line 13), however, don't specifically teach combining the two images into one image and buffering this combined image prior to display. Nagata teaches a system in which multiple images are combined for display (see abstract), similar to that of DiNicola and Ross, but further teaches a the steps of a mixer mixing the two images and then transmitting the combined image to a buffer prior to displaying the image (see column 2, line 15 through column 3, line 4 and in figures 1 and 2). It would have been obvious to one of ordinary skill in the art, having the teachings of DiNicola, Ross, and Nagata before him at the time the invention was made to modify the image combination system of DiNicola and Ross to include the combination of images followed by buffering the combined image, followed by display, as is done in Nagata. One would have been motivated to make such a combination because this allows for reuse of the displayed image for further additions to the display space (as would be the case in animation / video environments).

The examiner will now address the individual arguments and statements made by Appellant.

From page 16 of the Appeal Brief, from the second paragraph, the Appellant argues that “neither the “raster lines” of Nagata nor the "bit planes” of DiNicola may be analogized to images”.

The Examiner respectfully contends that Nagata specifically teaches a “multi-layer image mixing apparatus” that combines the images via two raster scans. DiNicola teaches for example “bit plane 24 can be designated to hold foreground objects, bit plane 26 can hold moving objects, while bit plane 28 holds background objects” where “image mixer 32 will combine the picture element data from each of the three bit planes" (see column 5, lines 32-56).

From page 16 of the Appeal Brief, from the third paragraph, the Appellant argues that DiNicola doesn’t teach either forwarding images in sequence to the display nor combining the images.

The Examiner respectfully contends that DiNicola teaches “a graphics display system that creates a composite image by merging multiple images from separate memory buffers” (see column 2, lines 25-29). Further see the example of the combining of images in column 5, lines 32-56.

However, Ross is relied upon for further teaching two distinct modes where in the “hardware cursor mode”/”overlay mode” each image is transmitted

in sequence to the display screen (ie: the on-screen memory is sent, then the cursor memory is sent overlaying the on-screen memory) and alternately in a "normal mode" each of the two images are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2). In other words Ross has two images in memory (1) the on-screen memory [18]; and (2) the cursor memory [20]; and the display memory [30], which is described as a frame buffer that stores the combined on-screen memory [18] and cursor memory [20] (see column 3, lines 50-57), as does a conventional display memory storing the image (one image frame) that is to be displayed on the screen.

From page 17 of the Appeal Brief, from the second and third paragraphs, the Appellant argues that DiNicola's bit planes do not represent "images" but rather merely elements of images.

The Examiner respectfully contends that DiNicola teaches for example "bit plane 24 can be designated to hold foreground objects, bit plane 26 can hold moving objects, while bit plane 28 holds background objects" where "image mixer 32 will combine the picture element data from each of the three bit planes" (see column 5, lines 32-56), clearly each of these bit planes are holding an image, that is part of a combined image when overlayed one upon the others.

From page 18 of the Appeal Brief, from the first paragraph, the Appellant argues that it would be improper to combine DiNicola with other art that describes buffering the composite image.

The Examiner respectfully contends that though there DiNicola "does not require" a frame buffer for the combination it doesn't preclude the buffering. Note that the bit planes 24, 26, 28, of DiNicola are described as “memory buffers” (see column 5, lines 4-16 and column 6, lines 18-20 and lines 30-32).

From page 18 of the Appeal Brief, from the second paragraph, the Appellant argues that in Ross the overlay mode represents the display of alternative images rather that a composite image.

The Examiner respectfully contends that Ross teaches in overlay mode that displaying the stored image in cursor memory over the selected portion of on screen memory (see column 6, lines 37-44).

> Ross *teaches two distinct modes:*
> *"hardware cursor mode"/"overlay mode" each image is transmitted in sequence to the display screen (ie: the on-screen memory is sent, then the cursor memory is sent overlaying the on-screen memory)*
> *"normal mode" each of the two images are buffered in a common on-screen memory (defined as a buffer) for display (see column 1, lines 36-41, column 3, lines 50-67, and column 4, lines 12-16 and in figure 2).*

From page 18 of the Appeal Brief, from the second paragraph, the Appellant argues that Ross obviously does not suggest buffering the combination prior to display. .

The Examiner respectfully contends that *Ross has two images in memory (1) the on-screen memory [18]; and (2) the cursor memory [20]; and the display memory [30], which is described as a frame buffer that stores the combined on-screen memory [18] and cursor memory [20] (see column 3, lines 50-57), as does a conventional display memory storing the image (one image frame) that is to be displayed on the screen.*

From page 18 of the Appeal Brief, from the third paragraph, the Appellant argues that Nagada does not transmit the combined image to a buffer prior to displaying the image.

The Examiner respectfully contends that Nagada teaches, in column 2, line 15 through column 3, lines 17 and figure 1, transmitting the combined image to image memory 40, where image memory 40 is a one frame memory (buffer) (see specifically column 2, lines 26-27). Image memory temporarily stores the image prior to display. In summary, the two images are combined by the mixer 30, the combined image is stored in the 1 frame image memory 40, and then displayed on the display monitor 130.

(11) Related Proceeding(s) Appendix

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Conferees:

/Dennis G. Bonshock/

Primary Examiner, Art Unit 2173

Dennis Bonshock
June 3, 2010
Primary Examiner AU 2173

/Kieu Vu/
Supervisory Patent Examiner, Art Unit 2173

/William L. Bashore/
Supervisory Patent Examiner, Art Unit 2175